

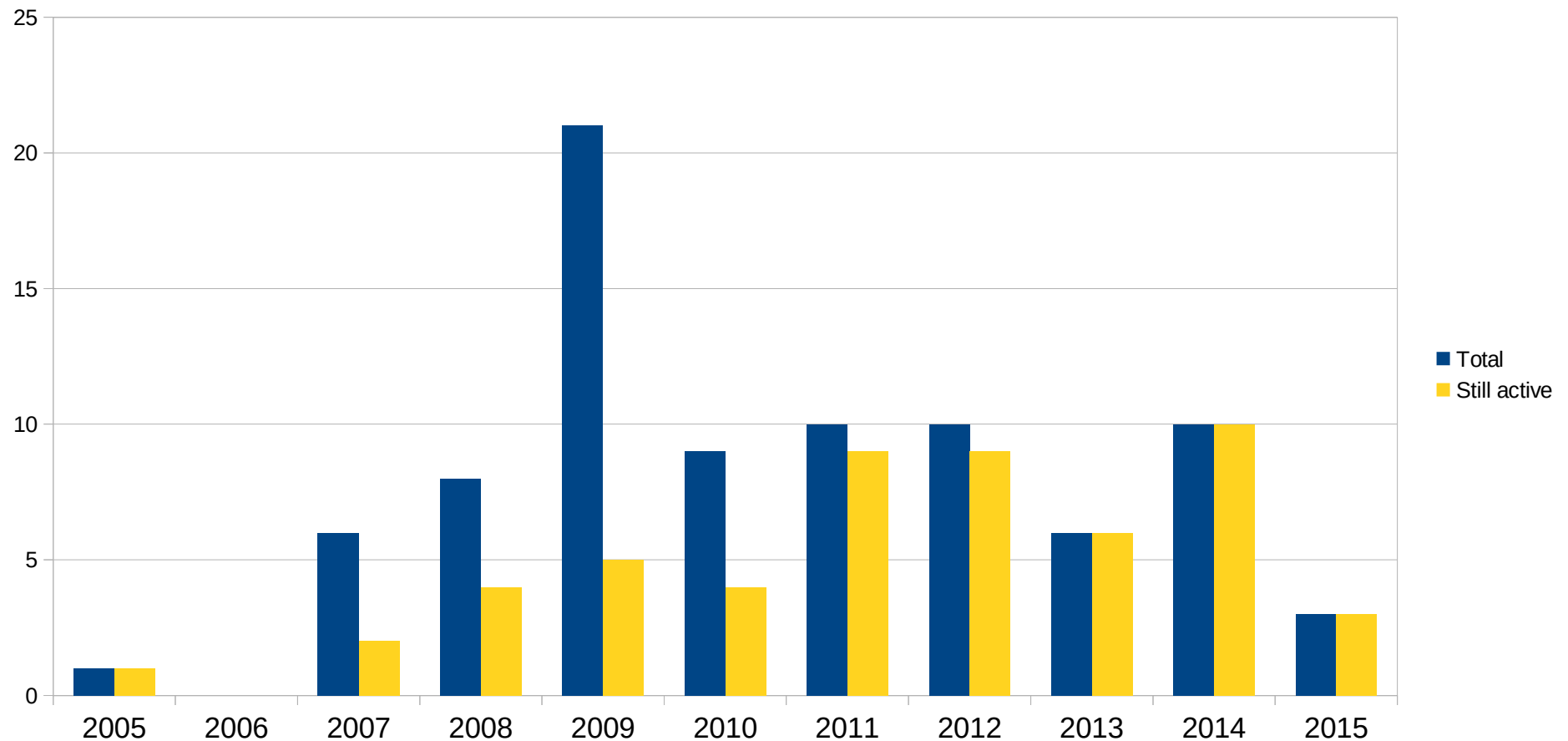
So you want to write an SMW extension

Yaron Koren
SMWCon Spring 2015
St. Louis, Missouri
May 7, 2015

SMW and its extensions

- SMW was created in 2005
- The first SMW-based extension was Semantic Forms, in 2007
- ~100 extensions have been created that require SMW, or that SMW requires

New SMW-related extensions per year



What do SMW-based extensions do?

- Provide **displays** of SMW data (e.g., Semantic Result Formats, Semantic Drilldown)
- Improve **editing** of SMW data (e.g., Semantic Forms, Halo)
- Allow for more data **storage** (e.g., Semantic Extra Special Properties, External Data)
- **Monitor** SMW data (e.g., Semantic Watchlist)
- Use SMW to store their **own data** (e.g., Semantic Tasks, Data Transfer)

So you want to create an SMW extension...

- First step: write the mailing list!
- Few people do this
- Maybe there's a better way to do what you're trying to do – or maybe it's already been done

Storing one's own data

Three main approaches:

- SMW special properties
- MediaWiki's page_props table
- Custom DB tables

1. SMW special properties

Example:

```
[ [Has background color::Green] ]
```

My suggestion: **avoid these entirely.**

Requires SMW, lots of translation needed, no nice display, no validation.

But... all my extension has is
special properties!

Change your extension, then?

Some “Semantic” extensions would be better off
not being “Semantic” at all.

I have been removing special properties

- Semantic Forms:

“Has default form”, “Page has default form”,
“Has alternate form” => **#defaultform**,
#formredlink

- Semantic drilldown:

“Has filter”, “Has input type”, etc. =>
#drilldowninfo

2. page_props table

Example:

```
{ {#page_display:background color=Green} }
```

...this might store the key/value pair
“PageDisplayBackground/Green” in page_props
for this page.

Great approach: parser function can set display
and validation; no other extensions required.

3. Custom DB tables

Example:

```
{{#page_display:background color=Green}}
```

...this would store data to the extension's DB table, called “page_display”.

Good solution when data is not page-specific, or not set via wikitext.

(i.e., not this example.)

Defining settings

- LocalSettings.php
- More settings are not better – less can be more.
- Resist the temptation to use settings due to indecivness.

Overusing settings

“Should this display be in bold or not? I can't decide...”

```
$wgMyExtensionDisplayResultsAsBold = true;
```

Problems with this:

- More code to maintain
- More documentation to read
- More decisions for admins
- Most users will not change the setting

Creating vs. modifying extensions

When possible, modify an extension – it's less work, and more people will use your code.

Major example: Semantic Result Formats. Why are people still creating one-format extensions?

Work within the system

Open source vs. proprietary? **Open source**

GPL vs. MIT vs. Apache vs. ? **GPL**

Wikimedia Gerrit vs. GitHub? **Gerrit**

Page on mediawiki.org vs. semantic-mediawiki.org vs. ... ? **mediawiki.org**

(These are Yaron's opinions only!)

A word on open source

“You want us to give away our software for free? It took us a lot of work to make!”

- Other people will make your software better
- Selling MediaWiki software is not a good business model
- (Consulting on it can be.)

